

[INICIAR SESIÓN](#)[NUESTROS PLANES](#)[TODOS LOS CURSOS](#)[FORMACIONES](#)[CURSOS](#)[PARA EMPRESAS](#)[ARTÍCULOS DE TECNOLOGÍA > DATA SCIENCE](#)

# Buscando tweets con Python



Yuri Matheus

09/12/2020

Cuando hacemos un análisis de datos, es muy común usar datos de una base de datos, archivos de texto, como .csv, archivos de log y otros. Además de estos, existen otros lugares donde podemos obtener datos para analizar.

Por ejemplo, cuando nos preguntamos sobre alguna tecnología. Podemos buscar en blogs, fórums, redes sociales o en la documentación de la propia tecnología. Todos estos sitios tienen informaciones que nos pueden ayudar con nuestra búsqueda.

Para análisis de datos también podemos buscar informaciones en sitios web en búsqueda de informaciones y también podemos utilizar la información de las redes sociales para entender mejor a nuestros clientes. Podemos utilizar estas informaciones para mejorar nuestros productos o sugerir el producto que mejor atienda las necesidades del cliente.

Existen varios sitios web y redes sociales que podemos utilizar para obtener informaciones, Twitter es un ejemplo. Por lo tanto necesitamos leer los tweets y obtener las informaciones, pero ¿pero imagina tener que leer cada tweet? Es decir, ¿necesitaríamos leer un tweet a la vez y hacer un análisis nosotros mismos de todos ellos?

Podemos hacerlo de esta manera, pero es seria bastante laborioso, ¿no crees? Quizas la mejor opción sea crear un **script** que nos busque los tweets con algun criterio especifico. Por ejemplo, con base en una palabra. Y luego, creamos otro script para clasificarlos.

Pero, ¿cómo conseguimos captar los tweets?

## Buscando tweets

Para buscar tweets, necesitamos comunicarnos con la [API de Twitter](#). Hay algunas bibliotecas en Python que podemos usar para esto. Una ampliamente utilizada es [Tweepy](#). Con ella logramos, entre otras cosas, buscar tweets.

Como estamos usando Python, una forma de instalar esta biblioteca es usar el pip. Por lo tanto: `pip install tweepy`.

```
yuri@yuri-laptop:~$ pip install tweepy
```

Ahora, para usarla en nuestro código, simplemente importamos la biblioteca:

```
import tweepy
```

Si queremos tener acceso a todas las funciones del Twitter, necesitamos tener una cuenta. Lo mismo ocurre cuando tenemos que crear una aplicación que acceda Twitter. Necesitamos "iniciar sesión" en nuestra aplicación para acceder a su API.

Entonces tenemos que [registrar nuestra aplicación en Twitter](#). Aquí necesitamos crear una nueva aplicación, es decir, una aplicación que se comunicará con nuestra cuenta, y con eso, tenemos acceso a las informaciones que necesitamos para "iniciar sesión" en nuestra aplicación.

En la **pestaña Claves y Token de Acceso (Keys and Access Token)**, tenemos las informaciones que usaremos para conectarnos a Twitter. Es decir, la clave del usuario (Consumer key), el código secreto del usuario (Consumer secret), los token de acceso (Access token) y su token secreto (Access token secret).

Todas estas informaciones son las llaves de acceso a nuestra cuenta de Twitter y deben ser guardadas y no olvidadas. Y claro, no podemos compartir estas informaciones. Dado que estas claves son strings muy extensos, las guardaremos en variables para que sea más fácil de manipularlas:

```
import tweepy
```

```
clave_usuario = 'xxxxxxx'  
secreto_usuario = 'xxxxxxx'
```

```
token_acceso = 'xxxxxxx'  
token_acceso_secreto = 'xxxxxxx'
```

Pero todavía no hemos autenticado nuestra aplicación Twitter, solo tomamos los credenciales. Como vamos a acceder a Tweepy, podemos decirle a tweepy que se encargue de nuestros datos de autenticación.

Entonces, podemos decirle a Tweepy que nos autentique usando nuestras claves de acceso:

#restante del codigo

```
autenticacion = tweepy.OAuthHandler(clave_usuario, secreto_usuario)
```

La clase **OAuthHandler** es responsable de realizar la autenticación en Twitter. Mira que en el constructor solo informamos la clave del usuario y su secreto. Pero, ¿y los tokens?

El siguiente paso es indicar cuáles son los tokens de nuestra autenticación. Es decir, podemos asignamos un valor al token de acceso:

#restante del codigo

```
autenticacion = tweepy.OAuthHandler(clave_consumidor, secreto_consumidor)  
autenticacion.set_access_token(token_acceso, token_acceso_secreto)
```

¡Genial! Ya tenemos todos los datos para autenticar. Así que vayamos al tweepy para conectar con la API del Twitter con estos datos de autenticación:

#restante del codigo

```
twitter = tweepy.API(autenticacion)
```

Ya estamos conectados con Twitter, ahora podemos buscar (search) nuestros tweets.

#restante del codigo

```
twitter.search()
```

Pero, ¿qué vamos a buscar? Es decir, cuál es nuestra consulta (q)? En la [Documentación de Twitter](#), dice que debemos informar un parámetro de búsqueda llamado q. Por ejemplo, si queremos buscar sobre los tweets de Alura o tweets que la mencionen, podemos informar su nombre como parámetro:

#restante del codigo

```
twitter.search(q='AluraOnline')
```

Ejecutemos el código y veamos si tenemos algún resultado:

```
(ml) yuri@yuri-laptop:~/Programas/Python/Buscando Datos$ python buscando_tweets.py
(ml) yuri@yuri-laptop:~/Programas/Python/Buscando Datos$
```

¿Dónde están nuestros tweets? Cuando usamos el método search, nos devuelve un objeto con el resultado de la búsqueda. Es decir, podemos asignar este resultado a una variable resultados, por ejemplo.

#restante del codigo

```
resultados = twitter.search(q='AluraOnline')
```

Para ver los tweets, podemos desplazarnos por los resultados y para cada tweet dentro de los resultados, imprimimos el autor, es decir, el usuario (user) quién hizo el tweet y el mensaje del tweet (text).

#restante do codigo

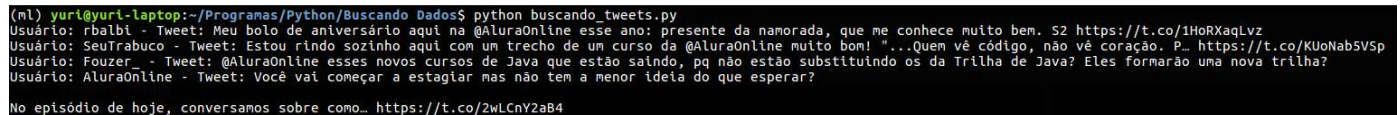
```
resultados = twitter.search(q='AluraOnline')
for tweet in resultados:
    print(f'Usuario: {tweet.user} - Tweet: {tweet.text}')
```

```
(ml) yuri@yuri-laptop:~/Programas/Python/Buscando Datos$ python buscando_tweets.py
Usuário: User(_api=<tweepy.api.API object at 0x7fdf0b754860>, _json={'id': 16679344, 'id_str': '16679344', 'name': 'Rafael Balbi', 'screen_name': 'rbalbi', 'location': 'Brazil', 'description': 'Pintista de executivo mas sou medieval.\nStreamer no @perdidosnoplay e @regradacasa. \nDesigner no @RpgNoticias \nDesign e Ilustração na https://t.co/VqnmOVzY0G', 'url': 'https://t.co/cXy8GF1CWD', 'entities': {'url': {'urls': [{'url': 'https://t.co/cXy8GF1CWD', 'expanded_url': 'http://www.pontosdeexperiencia.com.br', 'display_url': 'pontosdeexperiencia.com.br', 'indices': [0, 23]]}}, 'description': {'urls': [{'url': 'https://t.co/VqnmOVzY0G', 'expanded_url': 'http://Alura.com.br', 'display_
```

Imprimió el objeto de usuario, es decir, su id, nombre, ubicación y otras informaciones. Para una vista previa, sería mejor imprimir solo el nombre del usuario, es decir, el nombre que aparece en pantalla cuando accedemos a su perfil (screen\_name).

#restante del código

```
resultados = twitter.search(q='AluraOnline')
for tweet in resultados:
    print(f'Usuario: {tweet.user.screen_name} - Tweet: {tweet.text}')
```



```
(nl) yuri@yuri-laptop:~/Programas/Python/Buscando Datos$ python buscando_tweets.py
Usuário: rbalbi - Tweet: Meu bolo de aniversário aqui na @AluraOnline esse ano: presente da namorada, que me conhece muito bem. S2 https://t.co/1HoRXaQLvz
Usuário: SeuTrabuco - Tweet: Estou rindo sozinho aqui com um trecho de um curso da @AluraOnline muito bom! "...Quem vê código, não vê coração. P_ https://t.co/KUoNabSVSp
Usuário: Fouzer - Tweet: @AluraOnline esses novos cursos de Java que estão saindo, pq não estão substituindo os da Trilha de Java? Eles formarão uma nova trilha?
Usuário: AluraOnline - Tweet: Você vai começar a estagiar mas não tem a menor ideia do que esperar?
No episódio de hoje, conversamos sobre como_ https://t.co/2wLCnY2aB4
```

¡Ahora sí! Tenemos una visualización más clara de los tweets.

## Para saber más

Además de Tweepy, tenemos otras bibliotecas que podemos usar para comunicarnos con la API de Twitter. Una es el [Twython](#). Su uso es similar al Tweepy.

En este caso, como no usamos una cuenta pagada para buscar tweets, no tenemos acceso a todos los tweets publicados, sin embargo, conseguimos un buen volume de información para empezar un análisis.

Además de buscar tweets, estas bibliotecas nos permiten buscar usuarios, averiguar el número de seguidores e incluso escribir tweets con algún mensaje como si estuviéramos usando la página web o la aplicación.

Logramos capturar los tweets. Ahora podemos guardarlos en una lista o en un archivo y hacer nuestro análisis respectivo. Aquí en Alura, tenemos [cursos de Machine Learning](#) donde puedes aprender diferentes técnicas y algoritmos que puedes aplicar en tus análisis de datos.

Puedes leer también:

- [Cómo eliminar filas y columnas con Pandas en Python](#)
- [Trabajando con archivos y directorios en Python](#)
- [¿Cómo comparar objetos en Python?](#)

ARTÍCULOS DE TECNOLOGÍA > DATA SCIENCE

## En Alura encontrarás variados cursos sobre Data Science. ¡Comienza ahora!

**SEMESTRAL**

**US\$49,90**

un solo pago de US\$49,90

- ✓ 218 cursos
- ✓ Videos y actividades 100% en Español
- ✓ Certificado de participación
- ✓ Estudia las 24 horas, los 7 días de la semana
- ✓ Foro y comunidad exclusiva para resolver tus dudas
- ✓ Acceso a todo el contenido de la plataforma por 6 meses

**¡QUIERO EMPEZAR A ESTUDIAR!**

[Paga en moneda local en los siguientes países](#)

**ANUAL**

**US\$79,90**

un solo pago de US\$79,90

- ✓ 218 cursos
- ✓ Videos y actividades 100% en Español
- ✓ Certificado de participación
- ✓ Estudia las 24 horas, los 7 días de la semana
- ✓ Foro y comunidad exclusiva para resolver tus dudas
- ✓ Acceso a todo el contenido de la plataforma por 12 meses

**¡QUIERO EMPEZAR A ESTUDIAR!**

[Paga en moneda local en los siguientes países](#)

Acceso a todos  
los cursos

Estudia las 24 horas,  
dónde y cuándo quieras

Nuevos cursos  
cada semana

## NAVEGACIÓN

PLANES

INSTRUCTORES

BLOG

POLÍTICA DE PRIVACIDAD

TÉRMINOS DE USO

SOBRE NOSOTROS

PREGUNTAS FRECUENTES



## ¡CONTÁCTANOS!

¡QUIERO ENTRAR EN CONTACTO!

### BLOG

PROGRAMACIÓN

FRONT END

DATA SCIENCE

INNOVACIÓN Y GESTIÓN

DEVOPS

AOVS Sistemas de Informática S.A  
CNPJ 05.555.382/0001-33

### SÍGUENOS EN NUESTRAS REDES SOCIALES



### ALIADOS



En Alura somos unas de las Scale-Ups seleccionadas por Endeavor, programa de aceleración de las empresas que más crecen en el país.



Fuimos unas de las 7 startups seleccionadas por Google For Startups en participar del programa Growth Academy en 2021

POWERED BY

## CURSOS

### Cursos de Programación

Lógica de Programación | Java

### Cursos de Front End

HTML y CSS | JavaScript | React

### Cursos de Data Science

Data Science | Machine Learning | Excel | Base de Datos | Data Visualization | Estadística

### Cursos de DevOps

Docker | Linux

### Cursos de Innovación y Gestión

Productividad y Calidad de Vida | Transformación Ágil | Marketing Analytics |  
Liderazgo y Gestión de Equipos | Startups y Emprendimiento